# Synchro-Phasor Data

# Validation and Conditioning Project

## Phase 2 Report on

## Demonstration of the Prototype for Data Validation and Conditioning

*Prepared for the*
*Office of Electricity Delivery and Energy Reliability,*
*Transmission Reliability Program of the U.S. Department of Energy*
*Under Contract No.DE-AC03-76SF00098*

Prepared by:

Electric Power Group

Ken Martin
Jianzhong Mo
Sandeep Dua

Project Manager: Dejan J. Sobajic

**August 11, 2014**

## Acknowledgement

The work described in this report was coordinated by the Consortium for Electric Reliability Technology Solutions, and funded by the Office of Electricity Delivery and Energy Reliability, Transmission Reliability Program of the U.S. Department of Energy through a contract with Electric Power Group, LLC administered by the Lawrence Berkeley National Laboratory.

# Preface

Synchrophasor systems are being deployed for power system operations and control throughout the world. As the real time control center operations become more reliant on synchrophasors, it is essential that the data is correct and accurate to prevent errors in operation. Data needs to be validated to assure no errors have been introduced in communication and processing. It also needs to be conditioned with other comparisons to assure it is accurate. Validation and conditioning must be accessible to applications using the data so they can be used to support decisions in real-time to operations. The Department of Energy (DoE) has funded this project to develop and demonstrate a prototype tool for Phasor Data Validation and Conditioning in real-time (DE-AC02-05CH11231).

As part of Phase 2 task 2, this Report documents the prototype test results and its presentation to project sponsors and industry stakeholders.

# Table of Contents

# List of Figures & Tables

# Synchro-Phasor Data Validation and Conditioning Project

# Phase 2 Report: Prototype Demonstration

## 1. Introduction

The Synchrophasor Data Conditioning and Validation Project sponsored by the US department of Energy Consortium for Electric Reliability Technology Solutions (CERTS) program was started in December 2012. The project objectives are to develop, prototype, and test various methods for conditioning and validating real-time synchrophasor data. The project is divided into three phases.

- Phase 1: Conceptual Design and Prototype Development
- Phase 2: Prototype Demonstration
- Phase 3: Functional Specifications of the Data Validation System

In Phase 1 Electric Power Group, LLC (EPG) completed the design and prototype development to meet the data validation and conditioning requirements. These requirements have been developed by EPG based on surveys, literature research, and experience in working with customers.

This report covers Phase 2, Prototype Demonstration. Phase 2 consisted of four tasks:

- *Task 1: Develop Error Simulation Utility*
- *Task 2: Data Validation Prototype Demonstration*
- *Task 3: Summary Report*
- *Task 4: Review Meeting with Project Participants.*

In Phase 2 Task 1, EPG created an error simulation utility. The utility allows a user to generate various communication and measurement errors in a saved data stream to be able to observe that they are detected and alarmed. This greatly simplifies testing of the prototype. This was completed in May and used to test the PDVC prototype. It has also been used to demonstrate the prototype to users. This report covers the performance of the PDVC prototype and presents the test results. It also describes the prototype demonstration, summarizes results, and describes meetings with project participants.

# 2. Prototype Functionality

The PDVC prototype is required to detect a range of errors which were identified and tested in Phase 1. This list includes both the errors that were specifically listed in the RFP and those identified in the first two tasks of Phase 1. The prototype is modular consisting of six modules, with each module addressing a stage of error detection. Some of the errors may be detected at more than one stage. All modules report to the user interface which provides displays and statistics on detected errors.

The algorithm modules start checking data at the communication interface and proceed through user designated topological comparisons. The first three modules validate that data has been received without corruption and the last three condition data using flag and signal comparisons. The modules and the type of algorithms used in each module are listed below:

1. **Module 1- Communication Interface:** This module is designed to check for errors that may be introduced in the communications chain such as dropped bits, incorrect message frames, and CRC errors.

2. **Module 2 – Message Characteristics:** This module checks for message format errors such as length, destination address, type identification, and CRC16-check.

3. **Module 3 – Timestamps:** This module checks time tags for sequencing, data rates and transmission delays.

4. **Module 4 – Quality Flags:** This module utilizes all the flags available in the C37.118 standard to distinguish between good, bad, and uncertain measurements. Bad data is converted to NaN, suspect data is flagged, and all data is passed on to the next module for further processing.

5. **Module 5 – Data Characteristics and Self-Checking:** This module incorporates algorithms to check for unreasonably high or low values of voltage, current and frequency, data that is stale (not refreshing), and excessively noisy. Depending on severity, data that fails testing is declared bad and set to NaN or uncertain and flagged.

6. **Module 6 – Topology Checking:** The last module uses system topology to build algorithmic logic checking. For example, the sum of currents into a bus should be 0, and voltages at the same bus should be the same.

Raw data input to the PDVC can be from one or several PMUs. It must be in a single C37.118 formatted data stream. The PDVC performs the validation checks and creates an error flag for each measurement. The flag indicates whether the data passes all checks, does not pass all but may be usable, or is invalid and not to be used. The PDVC also creates a conditioned data stream where bad data is set to NaN (not a number), questionable data is flagged, and good data is unchanged. The user can indicate the conditions for setting the data as bad. Note that the PDVC converts all data to floating polar; this format conversion loses no accuracy or precision, but is better for error evaluation.

For output the user can choose to receive data in two different ways. The first presents the original data. The second presents a conditioned data stream where bad or suspect data is changed to NaN so the application can use data without examining a quality flag. The application must decide which data to use. In both cases, data that is unquestionably bad will be changed to NaN. This includes data that is filler for missing data, data with a bad CRC, and similar corruptions. The quality flags can be included with either output ways for further application specific evaluation. This overall algorithm process is shown in Figure 1 below.

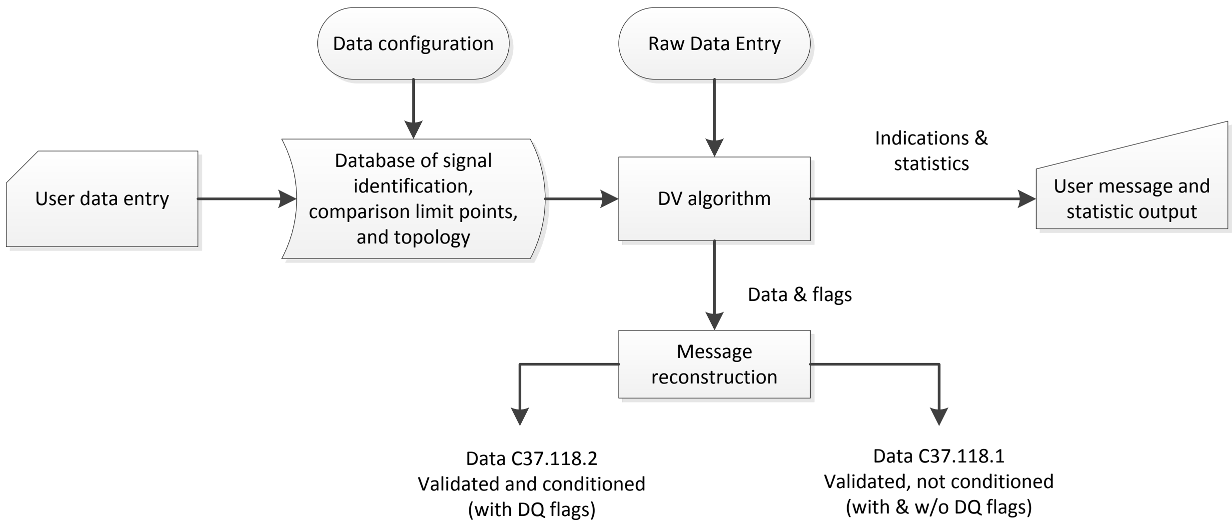



**Figure 1 Overall algorithm process**

# 3. Data Error Detection

## 3.1 Error categories

There are nine error type categories. These are listed in Table 1 below. This list includes both those identified in the RFP and in EPG research. Each category includes a number of specific errors that can be observed and used for conditioning. For example, data corruption is not an observable quality, but is something that is detected by looking at the CRC, the message characteristics, or other qualities. There are a number of ways that the corruption can occur and these are listed as possible causes. While the listed causes are the most common, there will certainly be additional circumstances that will cause errors.

**Table 1 Error type listing**

| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **1. Data corruption (data itself)** | - CRC error<br>- Message size error<br>- Message structure error<br>- Message timestamp incorrect | - Communication overrun<br>- Routing failure<br>- Communication bit error<br>- Data tampering or spoofing |
| **2. Intermittent communications, inconsistent data rates and latencies** | - Reporting rates change<br>- Messages out of order<br>- Transmission delays inconsistent | - Communication overrun<br>- Routing failure<br>- Data tampering or spoofing |
| **3. Loss of data from one or several PMUs** | - Data input fail—one or several messages | - Communication hardware failure<br>- Communication overrun<br>- Routing failure<br>- Loss of power to PMU<br>- PMU or PDC failure |
| **4. Loss of signals in a PMU** | - One or more measurements zeroed or random value<br>- Signal value missing from data output | - Signal input to PMU removed or failed<br>- PMU hardware or algorithm failure |
| **5. Offset in signal magnitude or phase** | - Signal magnitude error<br>- Signal phase error<br>- Measurement exceeds reasonable engineering limits | - Incorrect scaling<br>- Incorrect phasing<br>- Failure of an input to the PMU<br>- PMU hardware or algorithm failure<br>- Failure of the timing input<br>- Phasing does not match grid |
| **6. Corrupted and drifting signals in a PMU** | - Signal values corrupt (erroneous)<br>- Signal values drifting abnormally | - PMU hardware or algorithm failure<br>- PMU synchronization failure<br>- Time synchronization failure |
| **7. Corrupted and drifting time reference in one or several PMUs** | - Phase angle drift of PMU from others<br>- Frequency measurement error | - PMU time reference fail<br>- PMU synchronization fail<br>- GPS system failure |
| **8. Frozen or repeated (stale) measurements** | - Measurement always zero<br>- Measurement always nearly the same for a certain time period (not zero) | - PMU hardware or algorithm failure<br>- PDC communication failure<br>- Data tampering or spoofing |
| **9. Measurement incorrectly identified** | - Measurement does not track value<br>- Measurement does not make sense | - Data description mismatch<br>- Scaling error<br>- Data tampering or spoofing |

### 3.2 Testing procedure

In order to test the prototype to assure it will detect these errors, EPG developed an error simulation utility. This PMU Simulator is able to replay recorded data in CSV format, inject errors, and output error injected data in a C37.118.2 format data stream which feeds the PDVC prototype application. This Simulator is able to simulate data errors in real-time. The data flow of PMU Simulator is shown in Figure 2.

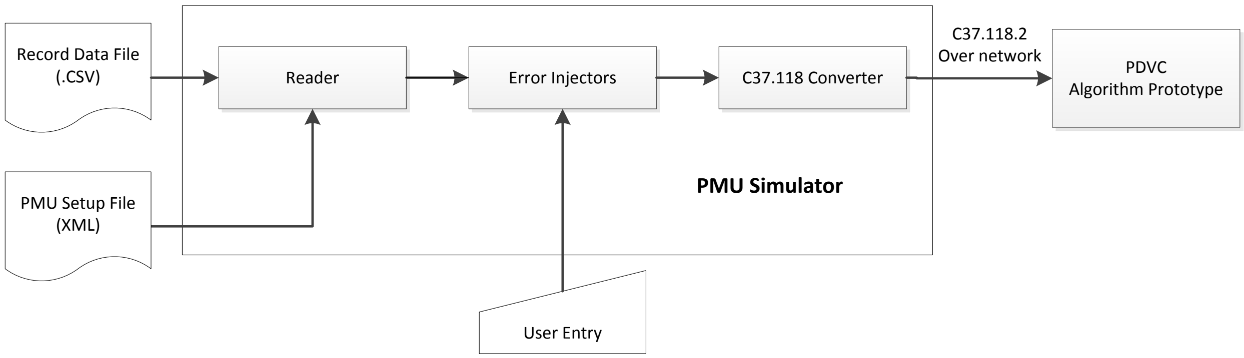



**Figure 2 PMU Simulator Data Flow to test the PDVC Prototype**

The simulator can use a file of real recorded data. It then takes this data and alters it to create real data errors. The tester then confirms that the errors are detected and correctly accounted (such as the number or duration of errors).

Testing included simulating the errors detailed under these 9 categories of errors. The PDVC correctly detected all errors and accurately counted all instances in the statistics. Section 4 of this report provides a description of the testing and sample screenshots of PDVC reporting.

## 4. PDVC tests and results

### 4.1 Overview

This section presents the test results. It is organized by error category as listed in section 3. *The prototype passed all tests successfully, so there is no pass-fail listing.* Sample screenshots for each section are given to show what the PDVC provides as output.

### 4.2 Data corruption

The data corruption category includes bit errors, incorrect protocol formats, timestamp errors, and check word errors. These can be caused by communication problems, hardware failures, and intrusions such as data spoofing. The first three PDVC algorithms monitor errors detected by the communication interface, check the protocol format, examine the time stamp and timing sequence, and check the check word. These checks cover the sources of data corruption. Figure -5 provide a series of screenshots that give examples of the PDVC detection and reporting data corruption type errors.

| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Data corruption** | - CRC error<br>- Message size error<br>- Message structure error<br>- Message timestamp incorrect | - Communication overrun<br>- Routing failure<br>- Communication bit error<br>- Data tampering or spoofing |

## CRC error detection

**Figure 3(a) PDVC receiving Good data**

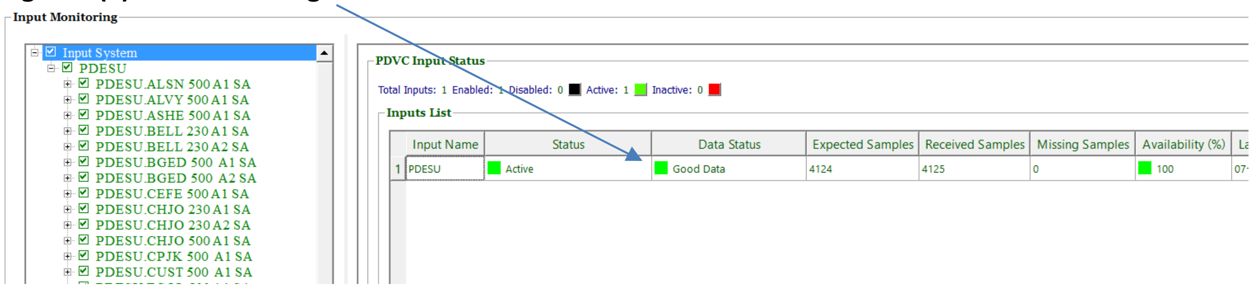
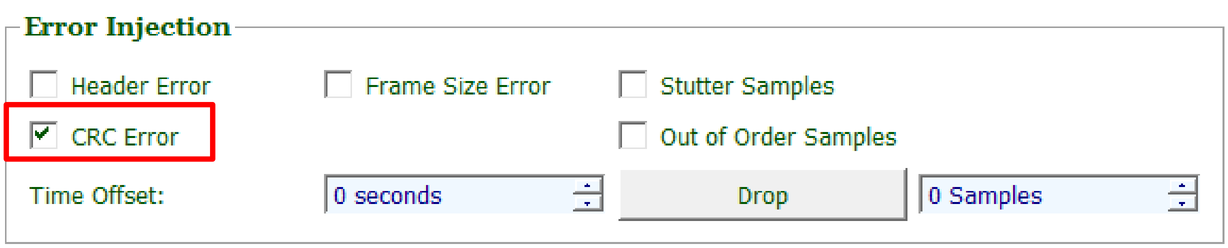
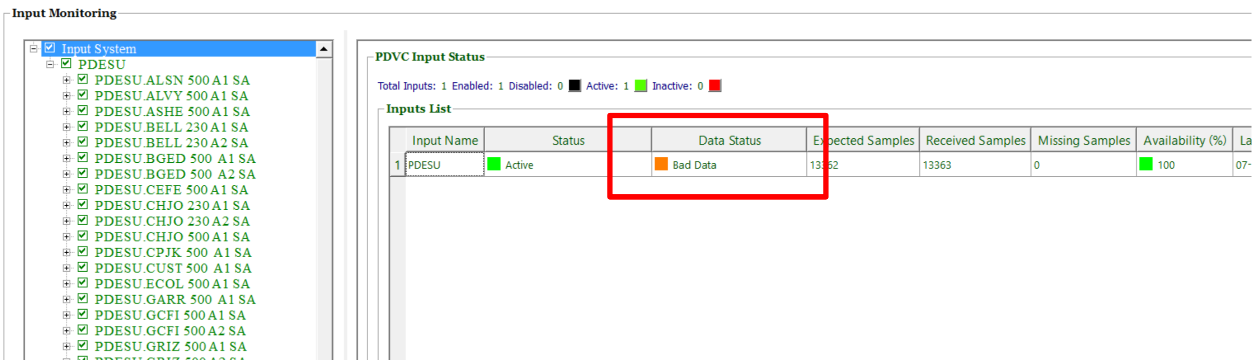


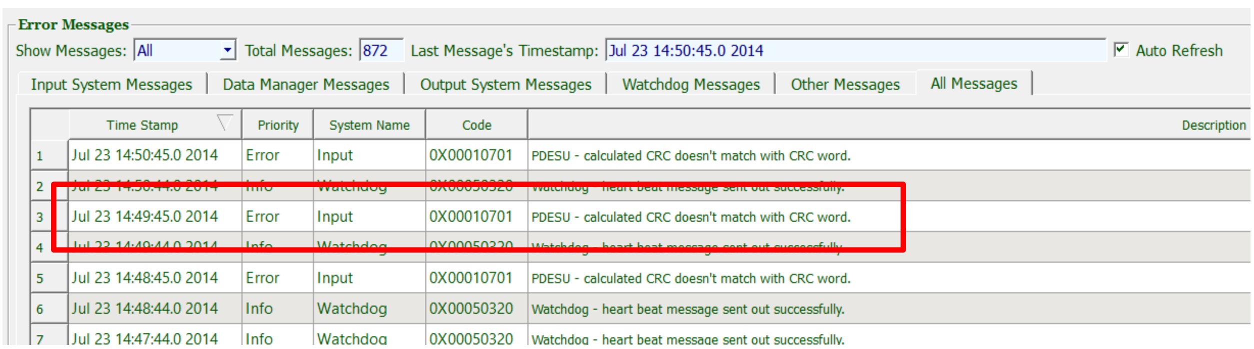

**Figure 3(b) CRC error injected in the data stream through Phasor Data Error Simulator Utility**



## Figure 3(c) PDVC Input Monitoring indicates problem with incoming stream




## Figure 3(d) PDVC logs reason for the error

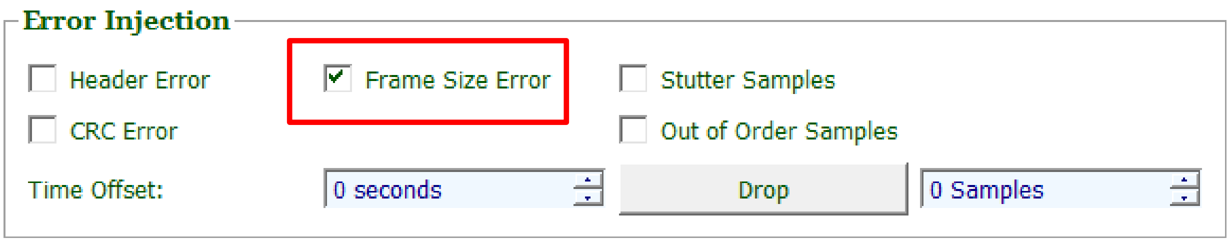



- *PDESU is the stream coming into PDVC from the Phasor Data Error Simulation Utility*

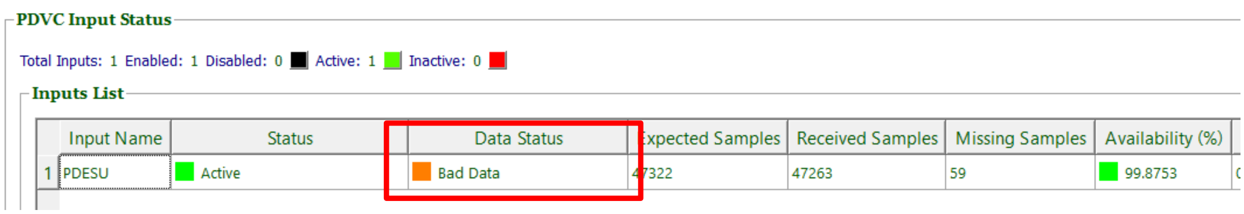
**Figure 3 (a-d) PDVC report for a CRC Error detected**

# Message size error

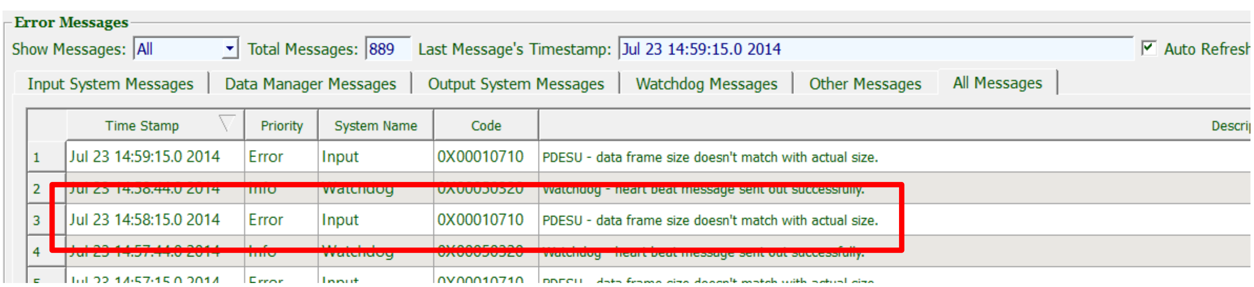
### Figure 4(a) Message / Frame size error injected through Phasor Data Error Simulator Utility




### Figure 4(b) PDVC Input Monitoring indicates problem with incoming stream




### Figure 4(c) PDVC logs reason for the error




- *PDESU is the stream coming into PDVC from the Phasor Data Error Simulation Utility*

**Figure 4 (a-c) PDVC report for a Message size Error detected**

## Message header error

**Figure 5 (a) Header error (Synch byte missing) injected through Phasor Data Error Simulator Utility**

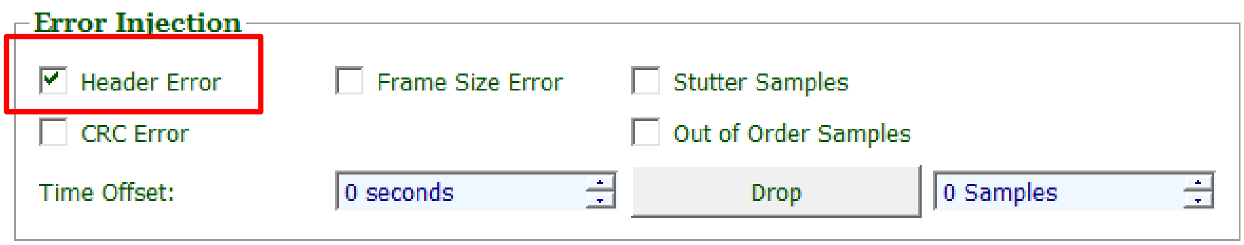



**Figure 5(b) PDVC logs reason for the error**

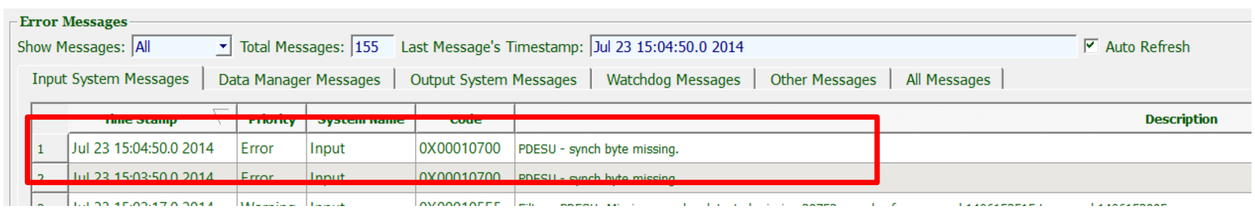



**Figure 5 (a-b) PDVC report for a Header Error detected**

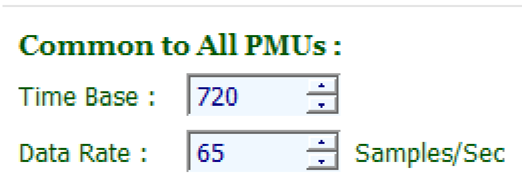
## 4.3 Intermittent communications, inconsistent data rates & latencies

Communication system problems can lead to input stream loss, packet loss, inconsistent latencies, packets out of order, data corruption, and similar problems. Since the communication is closely coupled with the production consumption of data, it can be difficult to determine whether a specific problem is due to communications, data sending equipment, or data receiving equipment. Troubleshooting will often require tapping into data streams at various points in the data chain to find the point where it fails. The error types described in this section are specific to communications. Figures 6-7 provide a series of screen shots that give examples of the PVDC error detection of these types of errors.

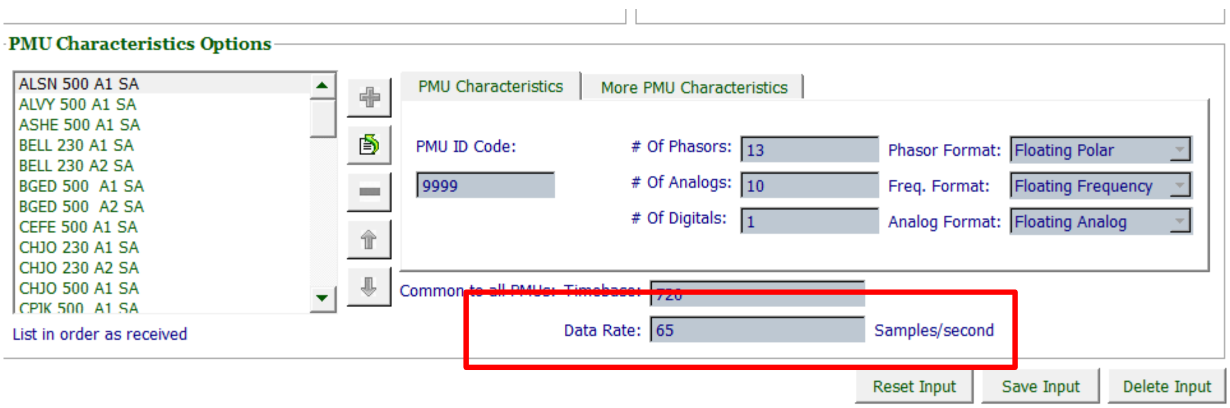
| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Intermittent communications, inconsistent data rates and latencies** | - Reporting rates change<br>- Messages out of order<br>- Transmission delays inconsistent | - Communication overrun<br>- Routing failure<br>- Data tampering or spoofing |

## Reporting rate change

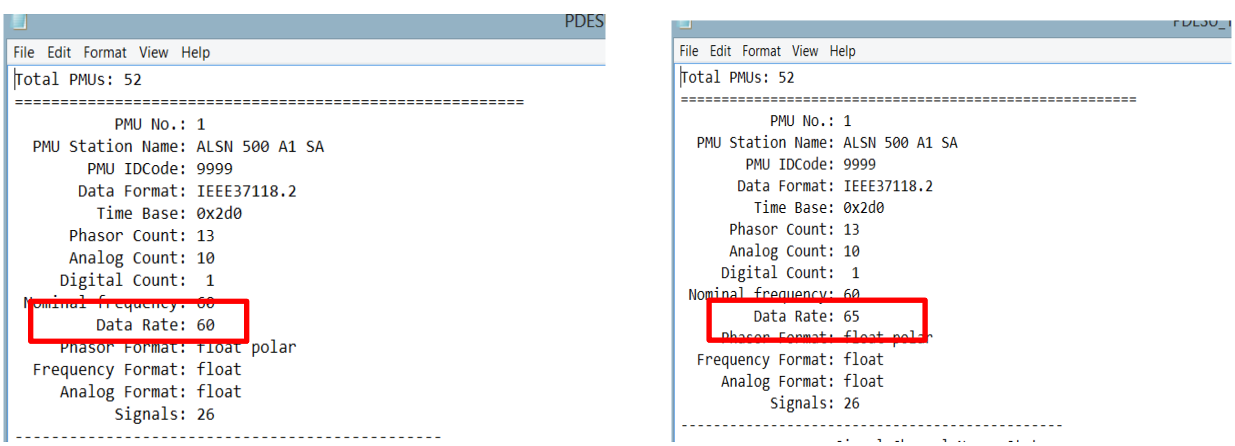
### Figure 6 (a) Reporting rate changed from 60 to 65 in the Phasor Data Error Simulation Utility




### Figure 6 (b) PDVC auto adjusts to the Data rate change using the updated configuration frame received




### Figure 6 (c) PDVC logs this change in the 'Config Frame Change Log'



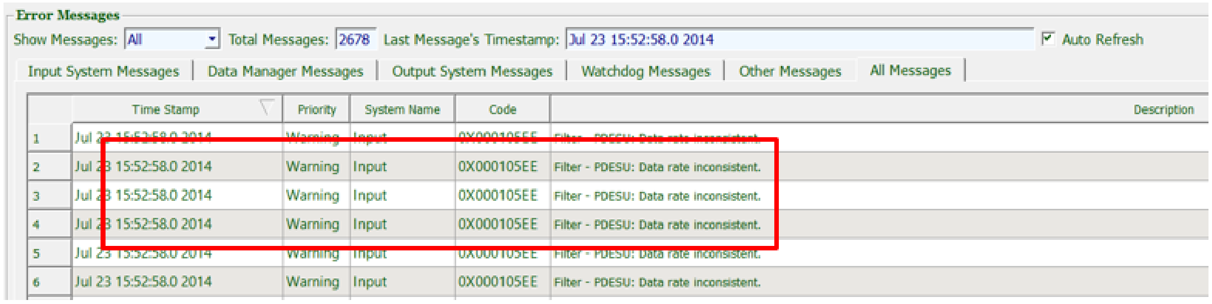



**Figure 6 (a-c) Data reporting rate change**

## Messages out of order

**Figure 7 (a) PDVC setting for Data sample internal inconsistency detection**

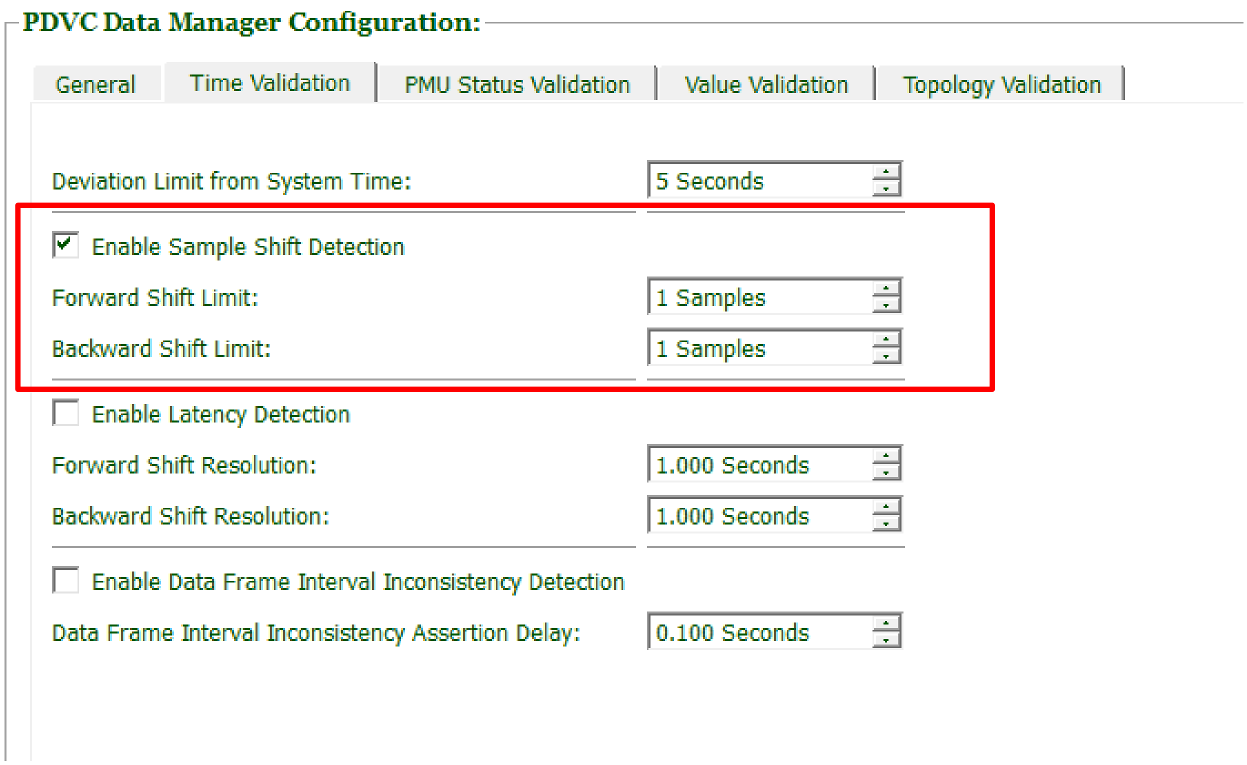


**Figure 7 (b) Out of Order Samples injected through Phasor Data Error Simulator Utility**

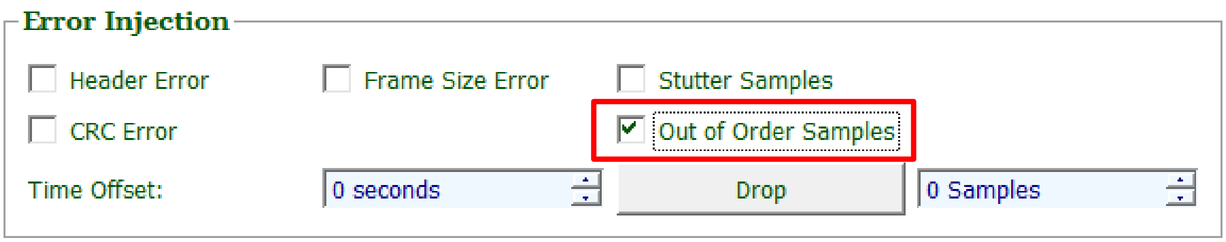
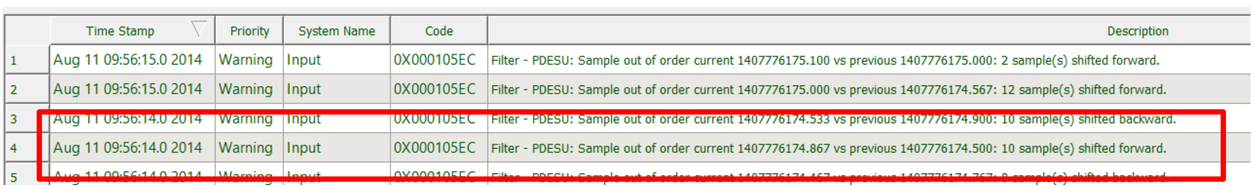



**Figure 7 (c) PDVC detects Out of order samples and corrects the order**



| | Time Stamp | Priority | System Name | Code | Description |
|---|---|---|---|---|---|
| 1 | Aug 11 09:56:15.0 2014 | Warning | Input | 0X000105EC | Filter - PDESU: Sample out of order current 1407776175.100 vs previous 1407776175.000: 2 sample(s) shifted forward. |
| 2 | Aug 11 09:56:15.0 2014 | Warning | Input | 0X000105EC | Filter - PDESU: Sample out of order current 1407776175.000 vs previous 1407776174.567: 12 sample(s) shifted forward. |
| 3 | Aug 11 09:56:14.0 2014 | Warning | Input | 0X000105EC | Filter - PDESU: Sample out of order current 1407776174.533 vs previous 1407776174.900: 10 sample(s) shifted backward. |
| 4 | Aug 11 09:56:14.0 2014 | Warning | Input | 0X000105EC | Filter - PDESU: Sample out of order current 1407776174.867 vs previous 1407776174.500: 10 sample(s) shifted forward. |
| 5 | Aug 11 09:56:14.0 2014 | Warning | Input | 0X000105EC | Filter - PDESU: Sample out of order current 1407776174.467 vs previous 1407776174.767: 9 sample(s) shifted backward. |

**Figure 7 (a-c) Out-of-order Data samples**

## 4.4 Loss of data from one or several PMUs

Data loss from a PMU can result from a number of problems. The most common is a communication system failure. However it could be the sending device, particularly when that device is secondary and receives data from some other device such as a PDC resending data it receives from a PMU. Figure 8 shows screenshots that provide an example of detection of this type of problem, simulated by removing a PMU from the input stream.

| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Loss of data from one or several PMUs** | - Data input fail—one or several messages | - Communication hardware failure<br>- Communication overrun<br>- Routing failure<br>- Loss of power to PMU<br>- PMU or PDC failure |

## Loss of PMU data

**Figure 8 (a) 2 PMUs are disabled in PDESU to simulate loss of PMU data**

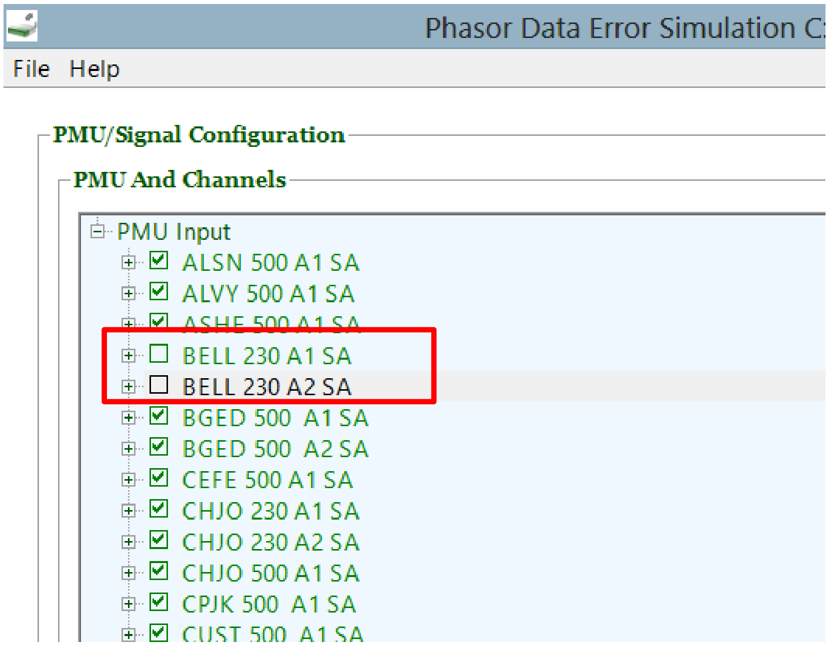
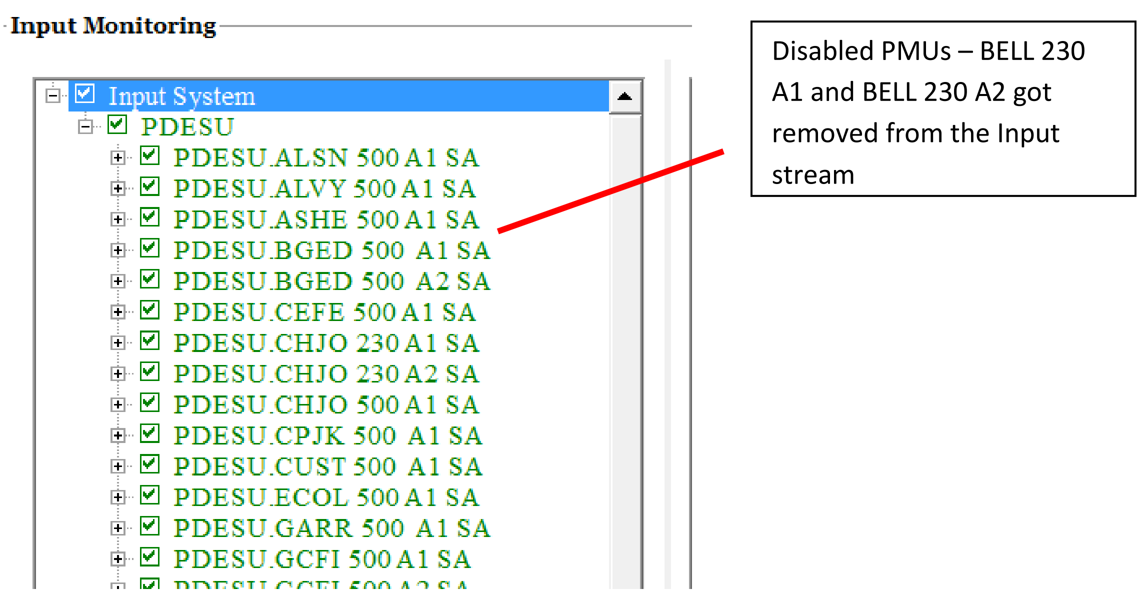



**Figure 8 (b) the PDVC removes these from the Incoming stream and disables them in the output stream**



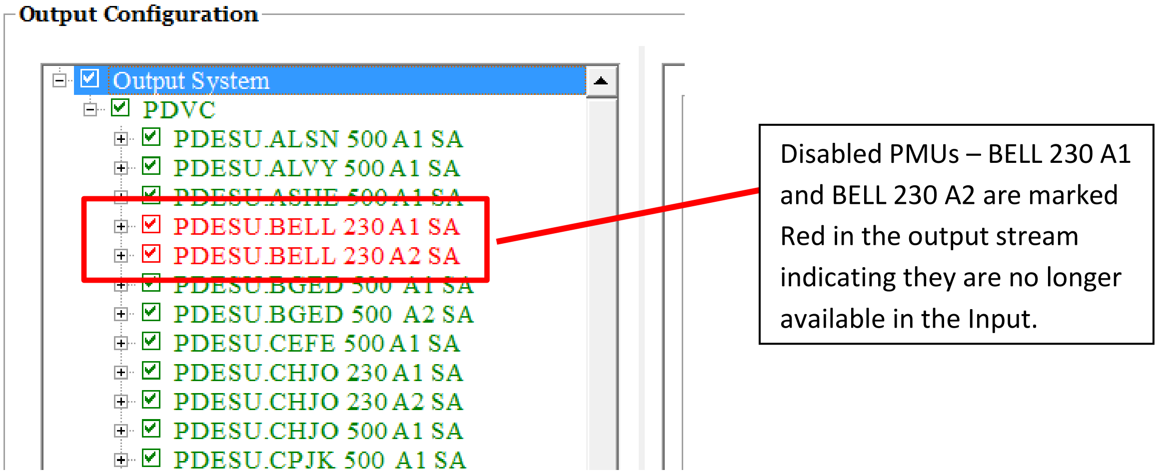



**Figure 8 (c) The PDVC logs indicating that the 2 PMUs and their signals have been removed from the Input stream**


**Figure 8 (a-c) Loss of data from one or more PMUs**

## 4.5 Loss of signals from a PMU

A PMU may continue to send the normal message stream but one or more of signal values may be missing or blanked. A missing signal value will only happen if the configuration is changed (that is, the data item is not in the stream). This indicates someone has changed the configuration or the sending device (PMU or PDC) has been corrupted forcing the change. Either case requires an investigation to determine the cause and the course of correction. The more likely loss of signal is when an input to the PMU fails so the measurement is incomplete or inaccurate. That can happen if one or more phases of a 3-phase signal are removed or become intermittent. This is easily detected if phasors for each phase are reported. If not, each phase contributes 33% to the positive sequence amplitude, so if that value drops by 1/3 or 2/3, loss-phases is a good guess. Loss of signal can also result from algorithm or hardware failures in the PMU, such as failure of the A/D converter for a single input phase. Figure 9 illustrates the PDVC detection of loss of signal from a PMU.

## Loss of Signals from a PMU

**Figure 9 (a) BGED Positive sequence Voltage is disabled in the PDESU Output**

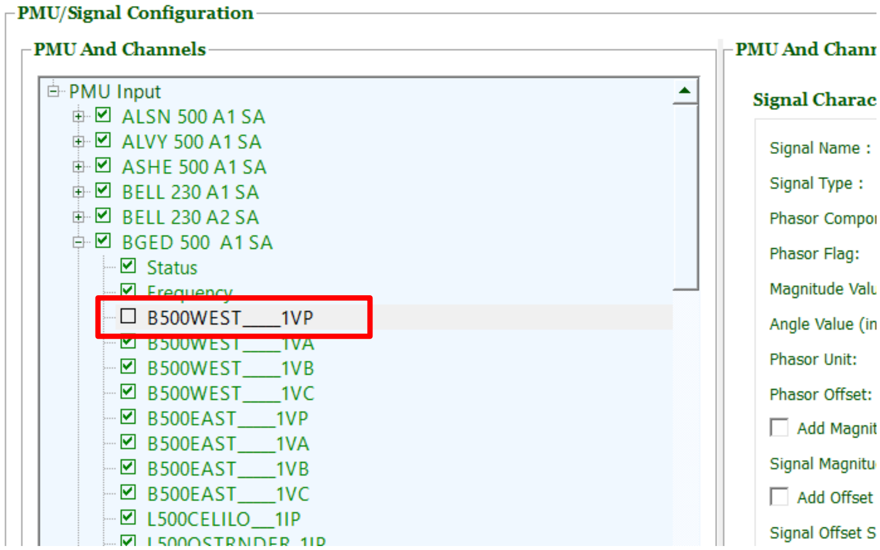
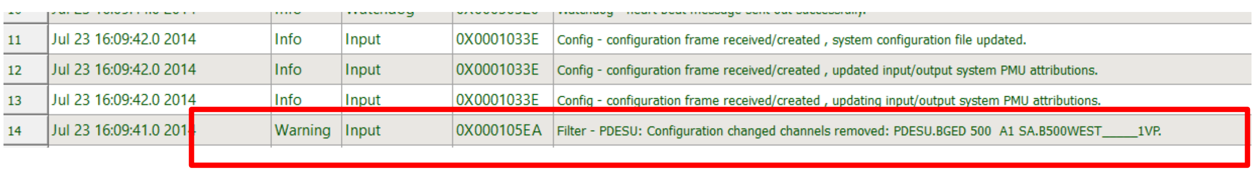



**Figure 9 (b) PDVC logs indicate that the BGED Positive sequence Voltage has been removed from the Input stream**



| | | | | | |
|---|---|---|---|---|---|
| 11 | Jul 23 16:09:42.0 2014 | Info | Input | 0X0001033E | Config - configuration frame received/created , system configuration file updated. |
| 12 | Jul 23 16:09:42.0 2014 | Info | Input | 0X0001033E | Config - configuration frame received/created , updated input/output system PMU attributions. |
| 13 | Jul 23 16:09:42.0 2014 | Info | Input | 0X0001033E | Config - configuration frame received/created , updating input/output system PMU attributions. |
| 14 | Jul 23 16:09:41.0 2014 | Warning | Input | 0X000105EA | Filter - PDESU: Configuration changed channels removed: PDESU.BGED 500 A1 SA.B500WEST_____1VP. |

**Figure 9 (a-b) Loss of a signal from a PMU**

## 4.6 Offset in signal magnitude or phase

A pure offset in the signal magnitude is rare since the AC signal is processed through a transform that will remove any DC offset from A/D conversion or other input problems. However scaling problems occur frequently through incorrect CT/PT ratios, instrument calibration, and incorrect signal identification. Conversely, scaling errors in phase angle are rare since they are derived from mathematical calculations, but offsets are common. The most frequent problems are timing signal unsynchronized to the UTC reference, incorrect phase identification on the PMU input, and regional phase differences. Unfortunately these types of problems cannot be detected just by examining the data. They require comparison between data items or with a reasonableness limit. For example, if two measurements report voltage for the same bus, they should remain within a small fraction of each other. The voltage measurement for a 500 kV bus should never be above 800 kV (except possibly momentarily). Comparisons and reasonableness limits can be entered by the user into the PDVC for this mode of validations. Figure 10 below illustrates using a reasonableness limit for frequency.

| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Offset in signal magnitude or phase** | - Signal magnitude error<br>- Signal phase error<br>- Measurement exceeds reasonable engineering limits | - Incorrect scaling<br>- Incorrect phasing<br>- Failure of an input to the PMU<br>- PMU hardware or algorithm failure<br>- Failure of the timing input<br>- Phasing does not match grid |

## Signal value validation failure

**Figure 10 (a) Low and High Passband set for CEFE Frequency signal in PDVC**

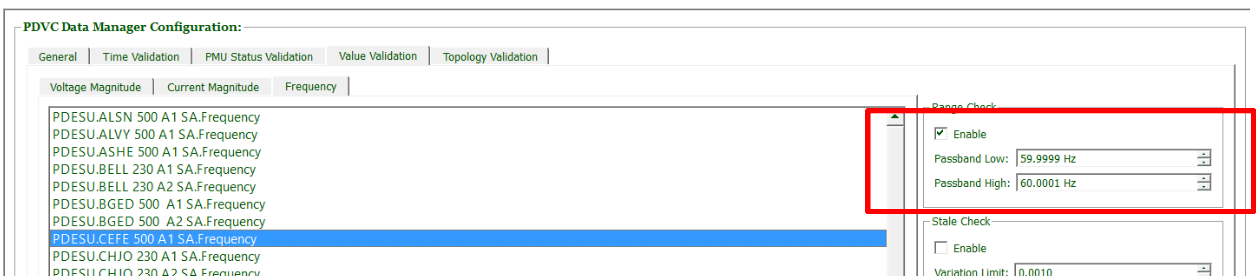
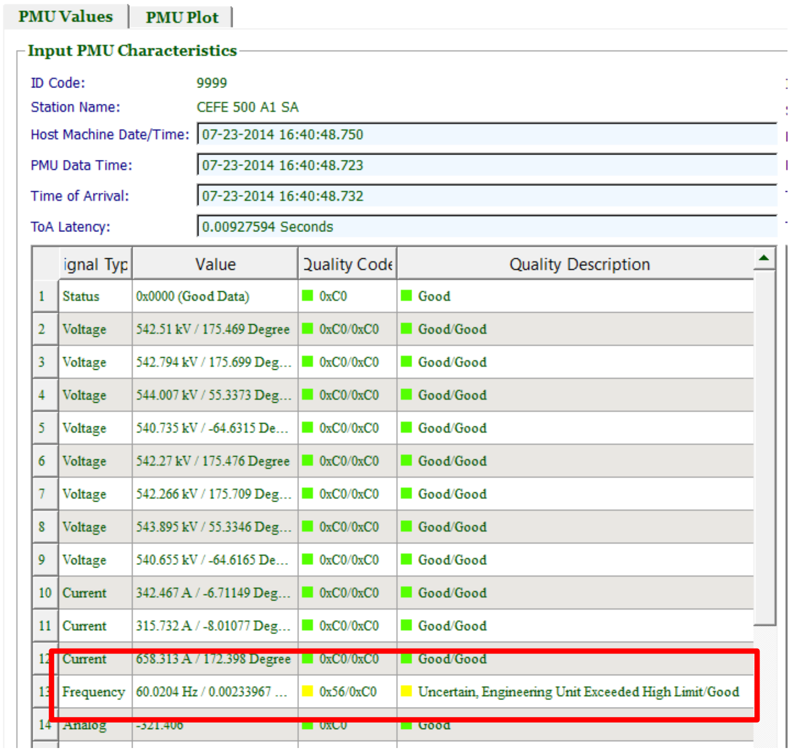



**Figure 10 (b) PVDC detects the Frequency signal is out of the passband limits and marks the data quality as Uncertain.**



PMU Values | PMU Plot

**Input PMU Characteristics**

ID Code: 9999
Station Name: CEFE 500 A1 SA
Host Machine Date/Time: 07-23-2014 16:40:48.750
PMU Data Time: 07-23-2014 16:40:48.723
Time of Arrival: 07-23-2014 16:40:48.732
ToA Latency: 0.00927594 Seconds

| | ignal Typ | Value | Quality Code | Quality Description |
|---|---|---|---|---|
| 1 | Status | 0x0000 (Good Data) | 0xC0 | Good |
| 2 | Voltage | 542.51 kV / 175.469 Degree | 0xC0/0xC0 | Good/Good |
| 3 | Voltage | 542.794 kV / 175.699 Deg... | 0xC0/0xC0 | Good/Good |
| 4 | Voltage | 544.007 kV / 55.3373 Deg... | 0xC0/0xC0 | Good/Good |
| 5 | Voltage | 540.735 kV / -64.6315 De... | 0xC0/0xC0 | Good/Good |
| 6 | Voltage | 542.27 kV / 175.476 Degree | 0xC0/0xC0 | Good/Good |
| 7 | Voltage | 542.266 kV / 175.709 Deg... | 0xC0/0xC0 | Good/Good |
| 8 | Voltage | 543.895 kV / 55.3346 Deg... | 0xC0/0xC0 | Good/Good |
| 9 | Voltage | 540.655 kV / -64.6165 De... | 0xC0/0xC0 | Good/Good |
| 10 | Current | 342.467 A / -6.71149 Deg... | 0xC0/0xC0 | Good/Good |
| 11 | Current | 315.732 A / -8.01077 Deg... | 0xC0/0xC0 | Good/Good |
| 12 | Current | 658.313 A / 172.398 Degree | 0xC0/0xC0 | Good/Good |
| 13 | Frequency | 60.0204 Hz / 0.00233967 ... | 0x56/0xC0 | Uncertain, Engineering Unit Exceeded High Limit/Good |
| 14 | Analog | -321.406 | 0xC0 | Good |

*'Uncertain, Engineering Unit Exceeded' is for Frequency and 'Good' is for Df/Dt.*

**Figure 10 (c) PDVC logging on signal measurement being out of range**

| | | | | | |
|---|---|---|---|---|---|
| 3 | Jul 23 16:40:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| 4 | Jul 23 16:40:05.0 2014 | Warning | Input | 0X000105E7 | Filter - Out of Range signal values exceeding high limit: PDESU.CEFE 500 A1 SA.Frequency. |
| 5 | Jul 23 16:39:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| 6 | Jul 23 16:39:05.0 2014 | Warning | Input | 0X000105E7 | Filter - Out of Range signal values exceeding high limit: PDESU.CEFE 500 A1 SA.Frequency. |
| 7 | Jul 23 16:38:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |

**Figure 10 (a-c) PDVC detecting, logging on measurement being out of range**

## 4.7 Corrupted or drifting signals in a PMU

Signals can be corrupted by a number of ways. There could be transmission errors that are not detected. There could be an algorithm failure. Data may be sent in integer form but decoded as floating point. It could be momentary corruptions like a data outlier or noise in a signal. Since the list is very long, the interpretation here is that the signals are simply in error. Likewise, a drift in signal is not precise, so it is here defined as a value that changes steadily but slowly in time, and not relating to the actual measurement.

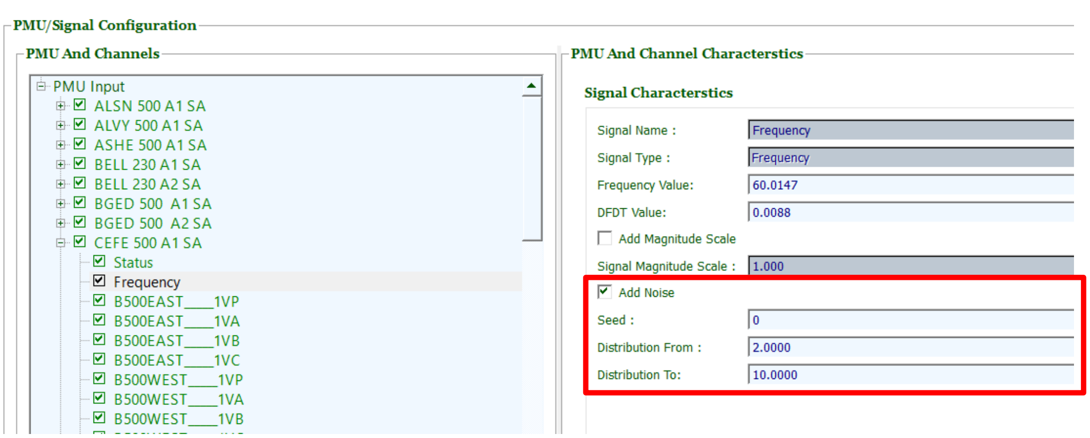
A drift in magnitude is most likely caused by a disconnected input that picks up induced signals which change as the coupling mechanism changes. This problem has not been observed by this author, but can happen. The comparisons described in section 4.6 are the best method to detect this type of error.

A drift in the measured phase angle is expected due to the fact it will rotate with the difference between the actual frequency and nominal frequency. However the difference between measured phase angles—which indicates the system or power factor angles, should not drift unless there is a change in the system. Since power system changes do occur, these drifts are difficult to separate from drifts caused by error conditions. Drifting phase angles are caused by synchronization errors which may occur in the timing system or the PMU. Synchronization errors should be detected by the clock and reported by the PMU. This will normally be the detection method. In the case that it is not, a drifting phase angle will result in phase angles abnormally large and reasonableness limits illustrated in section 4.6 can be used to detect the error. Comparison with signals that should report the same or similar angles is another good way to detect this type of problem. Figure 11 gives an example of detecting a corruption that appears as random noise on the signal.

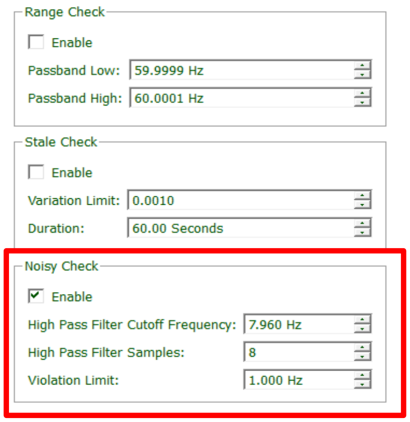
| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Corrupted and drifting signals in a PMU** | - Signal values corrupt (erroneous)<br>- Signal values drifting abnormally | - PMU hardware or algorithm failure<br>- PMU synchronization failure<br>- Time synchronization failure |

## Noisy Signal values

**Figure 11 (a) Noise injected in a Frequency signal using the Phasor Data Error Injection Utility**



## Figure 11 (b) PDVC detects Noise in signal based on configured parameters




## Figure 11 (c) PDVC Input monitoring displays the Data quality as Noisy

| | Signal Type | Value | Quality Code | Quality Descrip |
|---|---|---|---|---|
| 4 | Voltage | 543.774 kV / 148.583 Degree | 0xC0/0xC0 | Good/Good |
| 5 | Voltage | 540.482 kV / 28.6175 Degree | 0xC0/0xC0 | Good/Good |
| 6 | Voltage | 542.023 kV / -91.2765 Degree | 0xC0/0xC0 | Good/Good |
| 7 | Voltage | 542.016 kV / -91.0427 Degree | 0xC0/0xC0 | Good/Good |
| 8 | Voltage | 543.659 kV / 148.58 Degree | 0xC0/0xC0 | Good/Good |
| 9 | Voltage | 540.401 kV / 28.6319 Degree | 0xC0/0xC0 | Good/Good |
| 10 | Current | 344.773 A / 86.5876 Degree | 0xC0/0xC0 | Good/Good |
| 11 | Current | 311.579 A / 85.3328 Degree | 0xC0/0xC0 | Good/Good |
| 12 | Current | 656.459 A / -94.2848 Degree | 0xC0/0xC0 | Good/Good |
| 13 | Frequency | 66.4722 Hz / -0.00587448 Hz/s | 0x5C/0xC0 | Uncertain, Noisy/Good |
| 14 | Analog | -323.286 | 0xC0 | Good |

## Figure 11 (d) PDVC logging for detection of Noisy signals

| | Time Stamp | Priority | System Name | Code | |
|---|---|---|---|---|---|
| 1 | Jul 24 11:02:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| 2 | Jul 24 11:02:39.0 2014 | Warning | Input | 0X000105E8 | Filter - Noisy signal values: PDESU.CEFE 500 A1 SA.Frequency. |
| 3 | Jul 24 11:01:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| 4 | Jul 24 11:01:39.0 2014 | Warning | Input | 0X000105E8 | Filter - Noisy signal values: PDESU.CEFE 500 A1 SA.Frequency. |
| 5 | Jul 24 11:00:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |

**Figure 11(a-d) Example of corrupted data detection and logging**

## 4.8 Corrupted and drifting time reference in one or several PMUs

The time reference determines the phase angle measurement. A PMU will usually have a local oscillator that generates the internal conversion and synchronizing signals that is synchronized to an external reference. If the external reference is corrupted, the PMU will not be able to synchronize and the PMU oscillator will default to some frequency that will be close but not exactly the same as the reference should provide. The frequency difference causes an increasing phase difference which appears as a drifting phase angle. Similarly, if the reference signal itself loses sync with the primary UTC reference (usually through GPS), the measured phase angles will drift.

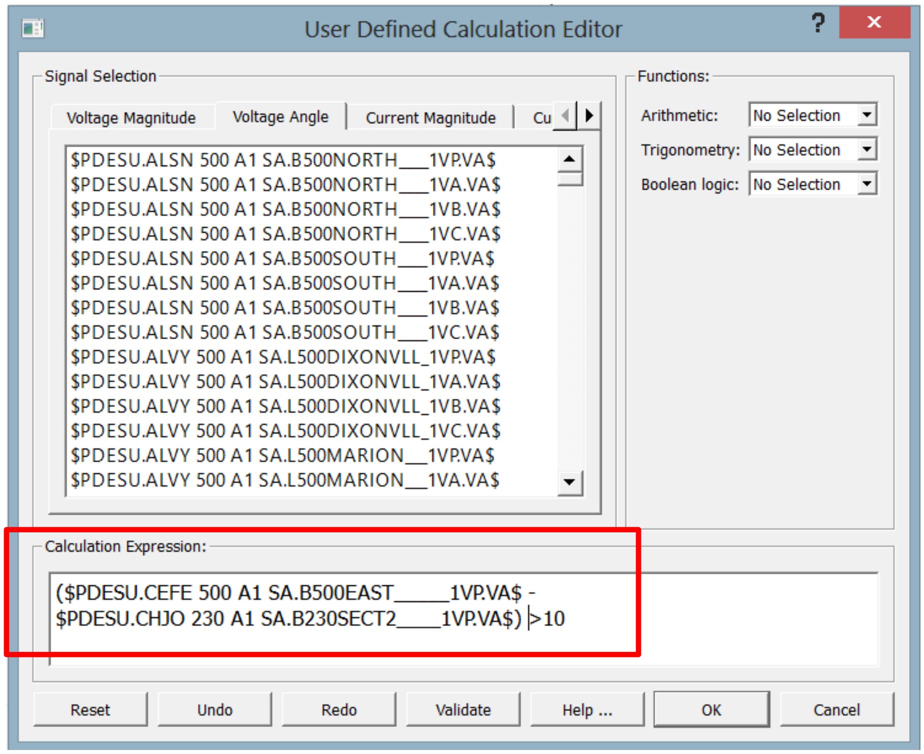
This type of error may be difficult to detect. If the difference is small or varies, the phase angle can appear like changes in power flow or an islanding event. If the time reference goes in and out of lock, the varying phase angle can look like load swings. Some of these problems can only be determined by careful examination of data when the angles look odd.

Generally when a PMU loses lock to the time reference, the frequency will go to a default value and stay there so the phase angle will drift in one direction. If a single PMU loses sync, the relative phase angle will drift from the others so relative angles will become bigger and/or smaller than normal, and the drift will go around the full circle in the same direction. If several PMUs lose sync, the same thing will happen but in more random directions since each one will be using a different local oscillator. Detection requires comparisons among PMUs to assure the phase angles stay within acceptable limits. When the angle difference gets large (positive or negative), an alarm indicates the angle is unacceptable and there may be a problem with the angle measurement. If the drift is fast enough, it could appear like a frequency error, but this would require such a large error that it is unlikely to trigger a frequency error. (For example, if a frequency limit alarm was set at 0.5 Hz, the measured frequency would have to be <59.5 Hz or >60.5 Hz. A frequency error of .5/60.5 = .83% is much larger than any common oscillator would produce and would result in a phase rotation of 180°/s.)

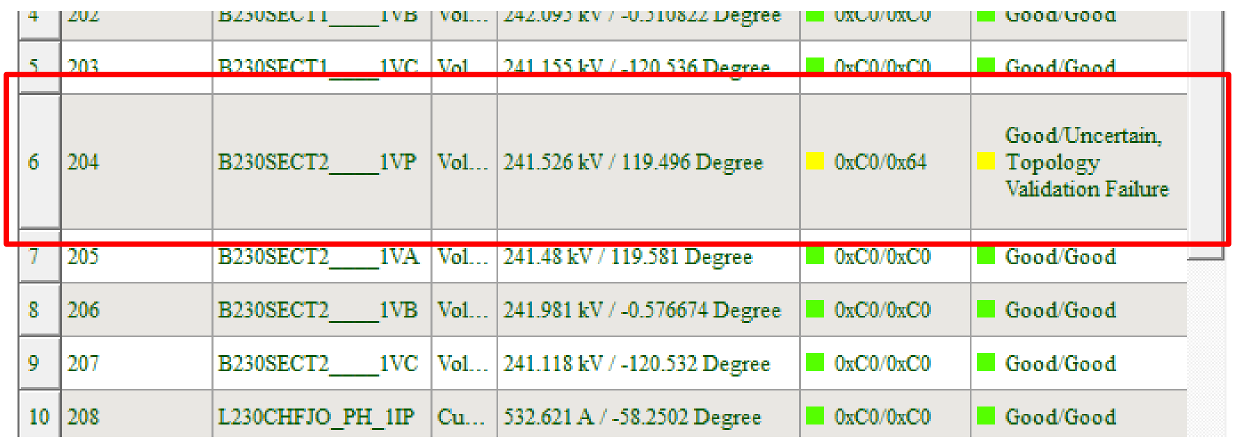
| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Corrupted and drifting time reference in one or several PMUs** | - Phase angle drift of PMU from others<br>- Frequency measurement error | - PMU time reference fail<br>- PMU synchronization fail<br>- GPS system failure |

## Topology validation to detect a Synchronization error

### Figure 12 (a) Topology check for voltage angles to detect sync failure




### Figure 12 (b) PDVC detects problem with Topology and flags the data accordingly



| | | | | | | |
|---|---|---|---|---|---|---|
| 4 | 202 | B230SECT1____1VB | Vol... | 242.095 kV / -0.310822 Degree | 0xC0/0xC0 | Good/Good |
| 5 | 203 | B230SECT1____1VC | Vol... | 241.155 kV / -120.536 Degree | 0xC0/0xC0 | Good/Good |
| 6 | 204 | B230SECT2____1VP | Vol... | 241.526 kV / 119.496 Degree | 0xC0/0x64 | Good/Uncertain, Topology Validation Failure |
| 7 | 205 | B230SECT2____1VA | Vol... | 241.48 kV / 119.581 Degree | 0xC0/0xC0 | Good/Good |
| 8 | 206 | B230SECT2____1VB | Vol... | 241.981 kV / -0.576674 Degree | 0xC0/0xC0 | Good/Good |
| 9 | 207 | B230SECT2____1VC | Vol... | 241.118 kV / -120.532 Degree | 0xC0/0xC0 | Good/Good |
| 10 | 208 | L230CHFJO_PH_1IP | Cu... | 532.621 A / -58.2502 Degree | 0xC0/0xC0 | Good/Good |

- The Data Quality for Voltage Magnitude is Good and state is uncertain due to topology error

**Figure 12 (c) Logging of errors related to Topology validation**

| ...me Stamp | Priority | System Name | Code | Description |
|---|---|---|---|---|
| ...:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| ...:31.0 2014 | Warning | Input | 0X000105E9 | Filter - Topology test failed: for signals: PDESU.CEFE 500 A1 SA.B500EAST_____1VP.VA, PDESU.CHJO 230 A1 SA.B230SECT2____1VP.VA. |
| ...:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| ...2:31.0 2014 | Warning | Input | 0X000105E9 | Filter - Topology test failed: for signals: PDESU.CEFE 500 A1 SA.B500EAST_____1VP.VA, PDESU.CHJO 230 A1 SA.B230SECT2____1VP.VA. |
| ...1:44.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |

**Figure 12(a-c) Example of using angle comparisons to detect a sync error**

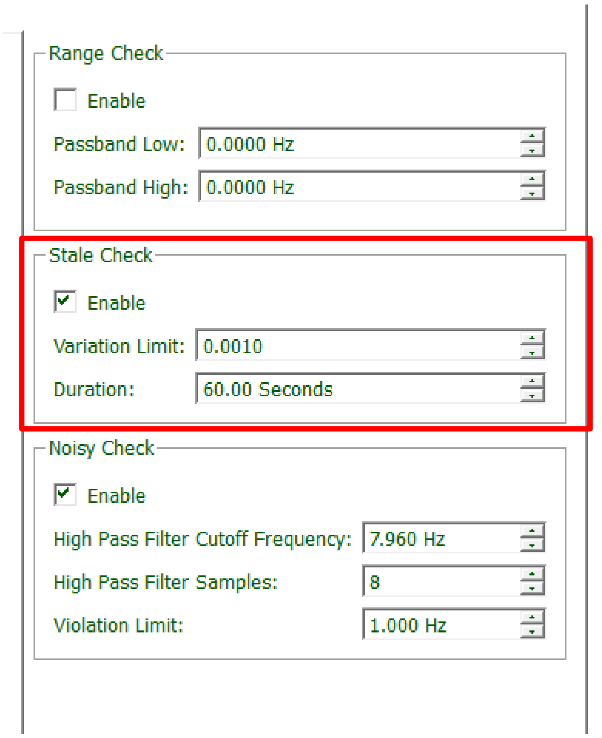
## 4.9 Frozen or repeated (stale) measurements

Frozen or stale measurements result from a failed input device that is passing the same values to the algorithm, an output that has stuck data, a failed IC that simply gives the same bit pattern, or from an intermediary device that is inserting a fixed value because it is not receiving new data. Inserting a constant value is also the simplest way to spoof a signal. Normally stuck data will be exactly the same each time but it could vary a little if a floating point conversion is being used. Real data will always have some variation, even if small. This detector allows the user to set a small variation band and a maximum time that the signal can remain in that band. If the signal remains within that band longer than the given time, an alarm indicates the signal is probably frozen. Note that the alarm needs to be intelligently set. The frequency can be within a 50 mHz band for hours, so limits for frozen frequency should probably be more like 8 mHz. An example is illustrated in Figure 13.

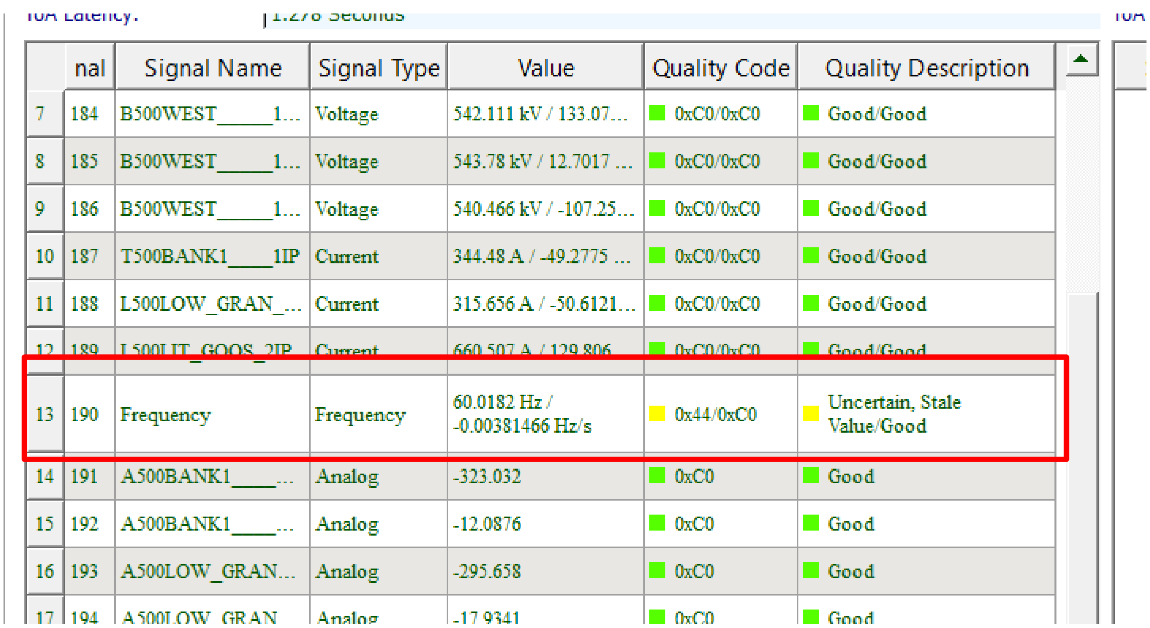
| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Frozen or repeated (stale) measurements** | - Measurement always zero<br>- Measurement always nearly the same for a certain time period (not zero) | - PMU hardware or algorithm failure<br>- PDC communication failure<br>- Data tampering or spoofing |

## Stale signal detection

### Figure 13 (a) PDVC State measurement detection filter turned on

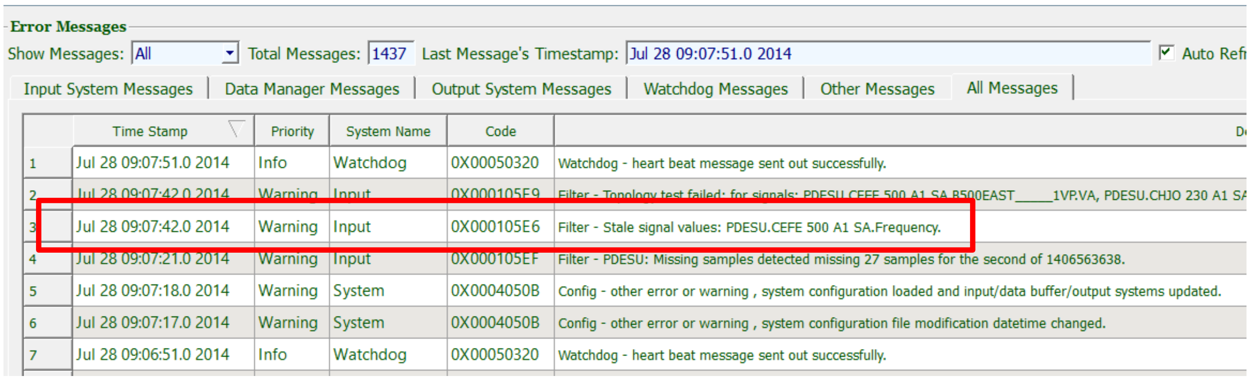



### Figure 13 (b) PDVC detects state signal values based on the configured parameters



| | nal | Signal Name | Signal Type | Value | Quality Code | Quality Description |
|---|---|---|---|---|---|---|
| 7 | 184 | B500WEST_____1... | Voltage | 542.111 kV / 133.07... | 0xC0/0xC0 | Good/Good |
| 8 | 185 | B500WEST_____1... | Voltage | 543.78 kV / 12.7017 ... | 0xC0/0xC0 | Good/Good |
| 9 | 186 | B500WEST_____1... | Voltage | 540.466 kV / -107.25... | 0xC0/0xC0 | Good/Good |
| 10 | 187 | T500BANK1_____1IP | Current | 344.48 A / -49.2775 ... | 0xC0/0xC0 | Good/Good |
| 11 | 188 | L500LOW_GRAN_... | Current | 315.656 A / -50.6121... | 0xC0/0xC0 | Good/Good |
| 12 | 189 | L500LIT_GOOS_2IP | Current | 660.507 A / 129.806 | 0xC0/0xC0 | Good/Good |
| 13 | 190 | Frequency | Frequency | 60.0182 Hz / -0.00381466 Hz/s | 0x44/0xC0 | Uncertain, Stale Value/Good |
| 14 | 191 | A500BANK1_____... | Analog | -323.032 | 0xC0 | Good |
| 15 | 192 | A500BANK1_____... | Analog | -12.0876 | 0xC0 | Good |
| 16 | 193 | A500LOW_GRAN... | Analog | -295.658 | 0xC0 | Good |
| 17 | 194 | A500LOW_GRAN | Analog | -17.9341 | 0xC0 | Good |

**Figure 13 (c) Logging of Stale signal data detection**




| | Time Stamp | Priority | System Name | Code | De |
|---|---|---|---|---|---|
| 1 | Jul 28 09:07:51.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |
| 2 | Jul 28 09:07:42.0 2014 | Warning | Input | 0X000105E9 | Filter - Topology test failed: for signals: PDESU.CEFE 500 A1 SA.B500EAST_____1VP.VA, PDESU.CHJO 230 A1 SA |
| 3 | Jul 28 09:07:42.0 2014 | Warning | Input | 0X000105E6 | Filter - Stale signal values: PDESU.CEFE 500 A1 SA.Frequency. |
| 4 | Jul 28 09:07:21.0 2014 | Warning | Input | 0X000105EF | Filter - PDESU: Missing samples detected missing 27 samples for the second of 1406563638. |
| 5 | Jul 28 09:07:18.0 2014 | Warning | System | 0X0004050B | Config - other error or warning , system configuration loaded and input/data buffer/output systems updated. |
| 6 | Jul 28 09:07:17.0 2014 | Warning | System | 0X0004050B | Config - other error or warning , system configuration file modification datetime changed. |
| 7 | Jul 28 09:06:51.0 2014 | Info | Watchdog | 0X00050320 | Watchdog - heart beat message sent out successfully. |

**Figure 13(a-c) Example of the stale data detection filter**

## 4.10 Measurement incorrectly identified

This is a surprisingly common problem but easily rectified. Measurements are made by devices like PMUs in substations by connecting them to wires that conduct electrical signals. In the typical substation there are hundreds of wire sets. Each has identification on terminal blocks but full identification requires relating to station drawings. After the PMU makes the measurement, it sends data in a block of numbers to the control center. Identification of the individual numbers requires a listing of the measurements and parsing the data block to match the list. Finally the measurements have to be matched with naming in a data storage system, like a database. The database may simply take names from the PMU system, or may have its own naming system that has to be cross referenced with PMU names. An error can occur in each one of these places, and the signal will be mis-identified.

A careful end-to-end data check should readily spot these problems. A field person checking the measurement directly with a control center person can usually clearly determine that the measurement is correctly identified. In some cases it may be necessary to cut out a signal to the PMU, such as if two sections of the same bus are being measured and one needs to determine which phasor represents each section. End-to-end checks are also good for spotting ratio errors.

Generally identification errors are not easily located by on-line monitoring. The limits monitor will alarm if the voltage is on the wrong bus or current is way out of expected values. Some topology monitoring will show when currents do not sum correctly or lime voltages do not match. But in many cases, the current and voltage will be within reasonable limits even when they are not the correct value. The recommended approach is to carefully validate the measurements when the system is installed and monitor for significant changes during operation (since reconfigurations can result in mis-identification). Also, set the limit checks to reasonable values and used topology checks where they can be applied. Since limit checks and topology checks have been illustrated in previous sections, no further examples are provided in this section.

| Error type | Error Details (Observed) | Possible causes |
|---|---|---|
| **Measurement incorrectly identified** | - Measurement does not track value<br>- Measurement does not make sense | - Data description mismatch<br>- Scaling error<br>- Data tampering or spoofing |

## 4.11 Conditioned Data from Validations

The PDVC conditions data by flagging it as “suspect” or “uncertain”, or declaring it “invalid”. If the data is received with a CRC error, then all of the data in that packet is unreliable and should be marked bad. If an intermediate PDC does not receive data for a particular time stamp, it has to put filler data in the packet as a placeholder, and it should be marked bad. So there are cases where the data is unquestionably bad. There are cases where data may be determined to be stale, but that only says it is probably bad, but not absolutely certain. In that case, the user may want a quality set or may want the data just set to bad.

The PDVC will output data in the same “raw” state as it was received or in a conditioned state. First note that all phasor and frequency data is converted to floating point and polar formats. All scale factors are applied. Any data unquestionably bad is set to NaN. With the first “raw” output option, all other data is output with the same value as received. With the second “cleaned” option, data which is questionable can be mapped to invalid at the users discretion. That way the cleaned data stream can be consumed by applications without having to compare with quality flags. However, the quality flags can be included with either output option, so can be used in either case.

The following Figure 14 illustrates how the user can define conditions that are mapped to invalidate the data.

**Figure 14 (a) PVDC options for mapping data quality**

- Include the calculated data quality flags in the Output stream
- Replace the Bad and/or Uncertain data with ‘NaN’


With both options selected, the downstream application (in this case ePDC) will receive the Data quality flags and bad/uncertain values replaced with ‘NaN’

**Figure 14 (b) Uncertain Frequency value replaced with 'NaN'**

| | | | | |
|---|---|---|---|---|
| 10 | 279 | T500BANK1____1IP | Current | 344.524 A / 95.9756 Degree |
| 11 | 280 | L500LOW_GRAN_2IP | Current | 309.661 A / 94.5684 Degree |
| 12 | 281 | L500LIT_GOOS_2IP | Current | 654.066 A / -84.9431 Degree |
| 13 | 282 | Frequency | Frequency | nan Hz / -0.00123359 Hz/s |
| 14 | 283 | A500BANK1____1MW | Analog | -323.075 |

**Figure 14 (c) Quality flags for all measurements as determined by PDVC**

*Quality code 22208 represents Uncertain data*

*Quality code 49344 represents Good data*

| | | | | |
|---|---|---|---|---|
| 20 | 2185 | QF_Frequency | Analog | 22208 |
| 21 | 2186 | QF_B500EAST_____ | Analog | 49344 |
| 22 | 2187 | QF_B500EAST_____ | Analog | 49344 |
| 23 | 2188 | QF_B500EAST_____ | Analog | 49344 |
| 24 | 2189 | QF_B500EAST_____ | Analog | 49344 |
| 25 | 2190 | QF_B500WEST_____ | Analog | 49344 |
| 26 | 2191 | QF_B500WEST_____ | Analog | 49344 |
| 27 | 2192 | QF_B500WEST_____ | Analog | 49344 |

**Figure 14(a-c) Example mapping flags to output and uncertain state to NaN**

# 5. Meeting with Project Participants

EPG has demonstrated the prototype to project participants at a meeting of WECC JSIS held on May, 20th-22nd 2014 in Salt Lake City, Utah. Subsequently, as requested by WECC, a WebEx was conducted on June 24th, 2014 to demonstrate the prototype. The WebEx was attended by 15 people from WECC, BPA, Peak RC, SRP, PG&E, SCE, Dominion, and DoE. Subsequently, EPG posted the error simulation utility and the PDVC prototype software for download by industry stakeholders from utilities and ISO's. In response to questions, EPG also indicated that commercial products named *enhanced* Phasor Data Simulator (*e*PDS) and *enhanced* Phasor Data Validator (*e*PDV) that build upon this research are in design and development and will be available later this year.

## 6. Summary

EPG developed an error simulation utility and prototype for the Phasor Data Validation and Conditioning project (PDVC) to meet the needs of synchrophasor data system users. This prototype was first tested with simulated data to prove that it detects errors of each type specified in the contract as well as others that EPG has encountered in its extensive work with these data systems. In this phase of development an error simulator was developed to test the prototype more extensively. This simulator can create specific errors in a data stream of real phasor measurement system data that has been recorded for this purpose. Since most of these errors occur only rarely, the only effective way to test and demonstrate error detection is to be able to create these errors on demand rather than wait for a real one to occur.

The prototype has been tested with real recorded data using the error simulator to inject errors. EPG demonstrated that each type of error was detected and flagged. In the case of providing a conditioned data output, EPG demonstrated that questionable values were replaced with NaN in cases where the user designated the change. As there are many possible combinations of errors and error indications, only a subset can be reasonably presented. This report discusses each category of errors and presents examples of each with screen shots of the setup, detection, and output indications.